European Commission

JRC TECHNICAL REPORTS

Interconnecting GRRASP with additional platforms and tools: A feasibility study

Authors
Luca Galbusera
Georgios Giannopoulos

2016

EUR 28073 EN

Joint Research Centre

European Commission
Joint Research Centre
Institute for the Protection and Security of the Citizen

Contact information
Georgios Giannopoulos
Address: Joint Research Centre, Via Enrico Fermi 2749, TP 186, 21027 Ispra (VA), Italy
E-mail: georgios.giannopoulos@jrc.ec.europa.eu
Tel.: +033278 6211
Fax: +033278 5469

JRC Science Hub
https://ec.europa.eu/jrc

Legal Notice
This publication is a Technical Report by the Joint Research Centre, the European Commission's in-house science service.
It aims to provide evidence-based scientific support to the European policy-making process. The scientific output expressed does not imply a policy position of the European Commission. Neither the European Commission nor any person acting on behalf of the Commission is responsible for the use which might be made of this publication.

All images © European Union 2016,

JRC 102547
EUR 28073

PDF ISBN 978-92-79-61295-4 ISSN 1831-9424 doi:10.2788/9389

Luxembourg: Publications Office of the European Union, 2016

© European Union, 2016

Abstract

As it has already been documents in several reports, the Geospatial Risk and Resilience Assessment Platform (GRRASP) under development ad the JRC is a World Wide Web oriented architecture bringing together geospatial technologies and computational tools towards the objective of supporting the analysis of critical infrastructures (CIs)
A key aspect of this platform is its capability to serve as the vehicle to interlink the analysis modules and tools that over the years have been developed by the scientific community towards a one-stop-shop for critical infrastructure risk and resilience analysis. The present report illustrates how the software architecture of GRRASP has been designed and is being exploited to support the integration of GRRASP with different projects related to the analysis of CIs.

Contents

List of Figures

Preface

The Geospatial Risk and Resilience Assessment Platform (GRRASP) developed by the JRC is a World Wide Web oriented architecture bringing together geospatial technologies and computational tools towards the objective of supporting the analysis of critical infrastructures (CIs).

A key aspect of GRRASP is its modular nature, pertaining to both core functionalities and the implemented scientific libraries. The present report illustrates how the software architecture of GRRASP has been designed and is being exploited to support the integration of different projects related to the analysis of CIs with GRRASP. This stems from the necessity to integrate a large number of tools and models that have been developed by the scientific community and remain scattered. As a consequence it is not possible to fully exploit the results of all the investment that has been performed in order to develop these tools.

In order to facilitate the integration of third party modules we have followed a specific process which is analysed as follows:

- In Chapter 1 we introduce architectural considerations on the GRRASP implementation and its relationship with the concept of data flow graphs;
- In Chapter 2 we analyze how some of the modules developed up to know are related to the mentioned graph architecture;
- In Chapter 3 we provide guidelines in order to support the simultaneous development of core components and scientific routines within GRRASP;
- In Chapter 4 we provide a short overview of one of the projects GRRASP is being interfaced with. (work in progress)

CHAPTER 1

GRRASP: architectural considerations

An inherent trait of the emerging critical infrastructure modelling and simulation methodologies is their variety in terms of representation granularity, sectoral focus and geographic extent. See for instance [1] for a recent review on modelling and simulation of interdependent CIs. Furthermore, in previous reports we discussed the necessity of developing analysis frameworks able to support multi-tiered analysis of CIs, e.g. by combining technological models and economic impact models to support a broad assessment of criticality and resilience of interconnected critical infrastructures. An example of application was provided for instance in [2], where we proposed a combined systems engineering and economic model for the analysis of a critical infrastructure network failure.

In the development process of the GRRASP platform, such aspects of the CI analysis landscape posed the challenge to propose an open architecture able to accommodate and merge heterogeneous analysis libraries tasks over the common ground of geographical data sources manipulation.

In order to support this objective within an open architecture, as mentioned in the preface, modularisation of both core functionalities and scientific routines plays a key role. This feature has a first advantage in allowing a smooth integration with a content management system chosen as a reference (Drupal, `https://www.drupal.org/`), which also allows the possibility to interface and merge components from GRRASP with a number of other libraries and frameworks, supported by a large (open-source) community. As a second point, modularity points towards the integration of the platform with different projects and geo-processing based analysis frameworks related to CIs.

As the platform evolves and expands through the creation and improvement modules, the mentioned objectives are targeted by exploiting a data-flow programming paradigm [3]. According to such paradigm, a program can be modelled as a directed graph whose topology is specified in terms of data flowing between operations. Operations are defined as (black-box) actions on input data streams which are invoked when inputs are available. The data-flow paradigm puts an emphasis on parallel processing and supports the implementation over large, decentralised systems.

Therefore, in this report we will illustrate how one of the formalisms typical of data-flow programming, namely data flow graphs, can serve the objective of fostering the development of the GRRASP platform and its integration with third-party analysis frameworks and contributed CI simulation methodologies.

1 Data flow graphs and diagrams

A data flow graph (DFG) is *"a graph model for computer programs that expresses possibilities for concurrent execution of program parts"* [4]. In particular, it is a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{v_1, \ldots, v_n\}$ is the set of nodes and $\mathcal{E} = \{e_1, \ldots, e_n\} \subseteq \mathcal{V} \times \mathcal{V}$ the set of edges. According to this formalism,

- nodes ("actors") represent *"operations (functions) and predicates to be applied to data objects"* [4] and generally endowed with input/output data ports;
- edges characterize data dependencies in terms of *"channels for data objects to move from a producing actor to a consuming actor"* [4].

A simple example of a DFG used to represent a mathematical formula is represented Figure 1.1.

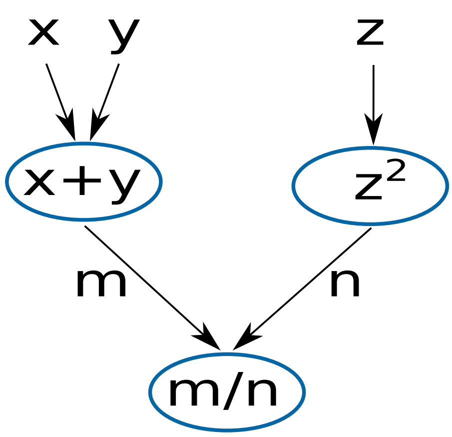

Figure 1.1: *DFG used to represent the formula* $\frac{x+y}{z^2}$.

Through the partial ordering of nodes imposed by the graph topology, this representation can be used to specify how operations are to be performed on data and transform them. Different rules can be specified on a given DFG for the structuring of the graph and the firing of the various operations on data. A number of DFG architectures have been proposed in the literature to date (http://www.iti.uni-stuttgart.de/~radetzki/Seminar06/11_report.pdf). Among the main formal models of DFGs, we find static data flow models, dynamic data flow models, and synchronous data flow models.

Among the features common to different types of DFGs, we find that action by a node is triggered by the presence of input data. A token mechanism is established to this end, in order to regulate the overall behaviour of a specified DFG. As DFGs also allow the presence of input ports, the formalism also accommodates the representation of interaction with inputs from the outer environment. Similarly, output ports can be specified.

As such, DFGs are a key formalism to represent data flow execution models, wherein *"all processing is performed by means of instructions that are applied to values. Instead of predefined scheduling, an instruction can be executed as soon as it has received all needed input data"* [5, 6].

Among the key features of the data flow execution model, we can mention [5, 7]:

- data dependencies equivalent to scheduling;
- single assignment of variables;
- locality of effect;
- freedom from side effects;
- lack of history sensitivity in procedures.

DFGs have an intrinsically modular architecture. They support the representation of computational concurrency. They allow distributed operation, the firing of various operations being triggered with no central control. DFGs are also apt to formal verification [8] and support performance improvement by means of transformations. According to [9], these include:

- **multi-rate expansion**, to convert multi-rate synchronous data flow graphs into single-rate synchronous data flow graphs;
- **retiming**, to rearrange the distribution of delay elements in the graph in order to maximize the throughput;
- **pipeling**, to add delay elements to the graph in order to optimize its iteration bound;
- **unfolding**, to duplicate nodes in order to enhance computational parallelism.

A DFG is *well behaved "if its activity terminates following each presentation of values"* [4]. As detailed in the same reference, the formalism allows the representation of conditionals and iterations and the specification of an *apply* actor.

As a complement to the DFG representation, we also shortly introduce the one of data flow diagram (DFD) to support the visualisation of data flow processing. Among the different notations available in the literature for DFDs, in this report we refer to the Yourdon/DeMarco notation [10], in which bubbles represent processes and arrow describe data flows, see Figure 1.2. This notation allows to represent data sources and sinks, as well.

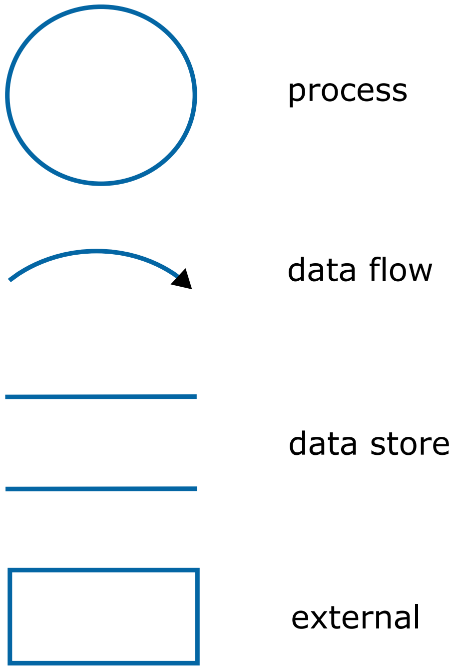

Figure 1.2: *DFD elements (representation according to the Yourdon/DeMarco notation).*

1.1 Relationships between GRRASP and DFGs

The DFG representation is able to accommodate different computational architectures and processes, and it is proposed as a reference in a number of implementations of scientific architectures today. An instance is provided by Google's TensorFlow™[1]. This is an open source software library for numerical computation based on data flow graphs and supporting a flexible deployment of computation over a variety of platforms and devices.

GRRASP is an open-source, client-server architecture. It is articulated in core functionalities, related to the management of the data flows and interaction with geographic data sources, and scientific modules. As we will discuss further in the rest of this report, the functionality of one such module can be interpreted as a DFG in itself, with source data generally coming from geographic sources and the user inputs, plus occasionally some non-geographic data souces.

Here we outline some of the advantages related to the exploitation of a DFG

[1] https://www.tensorflow.org/

model in GRRASP.

- **Hybrid deployment of DFG nodes**: a computational task expressed in GRRASP by a DFG can include nodes representing processing both by the server or the client; an example is provided by the case of a server-side simulation preceded by data manipulation and pre-filtering by the user, which could take place in the browser.
- **Multiple GRRASP instances**: GRRASP could be installed in multiple instances and, in perspective, allow data sharingg and distributed simulation processes still within the DFG formalism.
- **Tiered analysis**: the requirement for layered analysis typical of CI science could be supported by the structured concatenation of DFGs according to a data exchange formalism.
- **Concurrent and distributed geo-processing**: the inherent parallelism allowed by a data-flow programming model can be a key advantage in the processing of large amounts of geographic data, elaborated by means of formalised DFGs or DFG chains, with the possibility to support periodic or batch processing options.
- **Integration with existing DFG-based computational architectures**: the example of Google's TensorFlow™ seems a significant case and could be relevant, for instance, in order to apply artificial intelligence methods (e.g. to target image analysis and feature detection and comparison) [11] adaptively with respect to the available hardware, for a specific GRRASP installation.

CHAPTER 2

Scientific modules and tiered analysis

In this chapter we illustrate how scientific modules incorporated into a GRRASP installation relate to the concept of DFG. In particular, we will explain how a scientific module can be interpreted as a DFG, generally involving processing both by the server and the client, plus interaction with the end user. We will also observe how different scientific libraries share the need for common functionalities, which provides some guidelines for a structured approach to core and library development. We will further address this topic in Chapter 3. Finally, we will discuss how model tiering, involving multiple modules or instances of the same module, can be formulated by embedding the DFGs associated to different scientific modules into larger DFGs.

1 DFG interpretation of scientific modules in GRRASP

Our discussion of the representation of scientific modules in GRRASP in terms of DFGs focuses here on two cases: the network analysis module and the input-output inoperability module.

1.1 Network analysis module

This module implements topological analysis of networks in terms of graph metrics. The implementation of the module includes a number of metrics relevant to the critical infrastructure analysis objective.

Description of the computational process

The starting point for the calculation of such metrics is the selection of a network composed of nodes and edges. Conventionally, we assume that these data are provided to the client by a WFS service provider (e.g. GeoServer) in terms of two structured layers (i.e. layers containing graph connectivity information). Once the data layers are available to the client, the user performs a geographical selection over the nodes and edges layers, leading to the specification of a selected layer embedding jointly node and edge information. The process continues with the user specifying the metrics to be calculated, possibly together with further attributes relevant to the specific case under analysis. Finally, the computation of the metrics is requested to the server and takes place, returning the results (metrics computation output or error message) to the client. Depending on the situation, the output may be visualised on the map or displayed numerically.

DFG representation

In Fig. 2.1 we report a representation of the DFG describing the computational sequence described above. The network nodes and network edges blocks represent

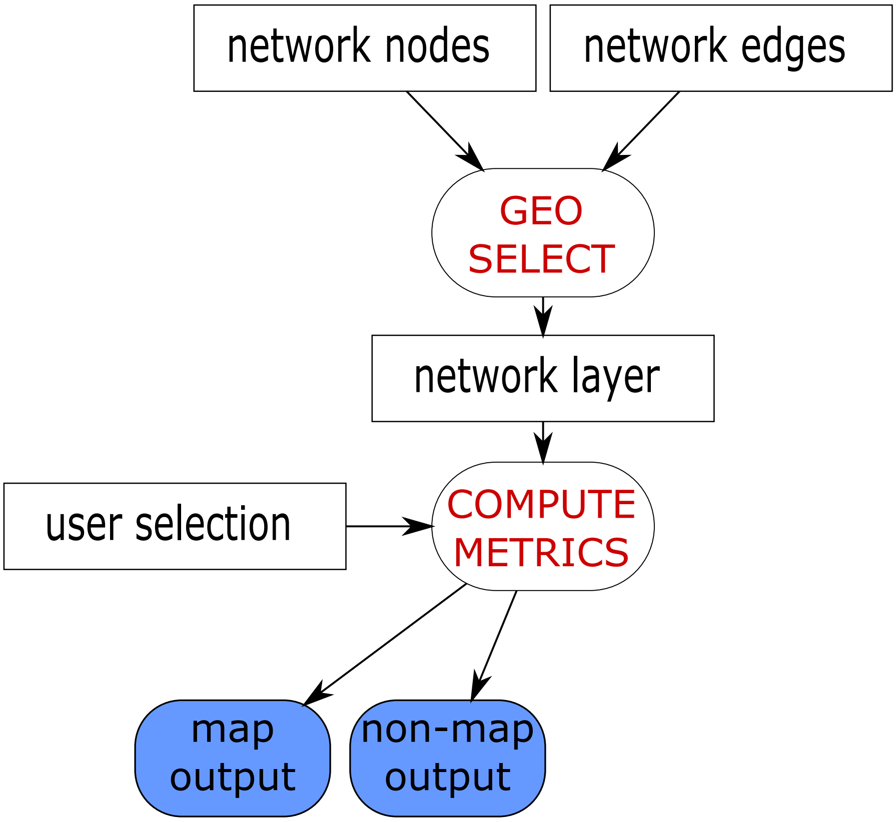

Figure 2.1: *DFG representation of the network analysis module.*

external nodes providing the geographical data. In our example implementation, these are provided by GeoServer by WFS services. The DFG is constructed starting from these layers through a sequence of operations consisting in a geographical selection (*GEO SELECT*) and the computation of the respective metrics (*COMPUTE METRICS*), subject to the specification of the user preferences (e.g. choice of the metrics and further options) by the end-user. The output of the metrics computation is structured, since some metrics admit map representation (e.g. node/edge-wise metrics) and some other not (e.g. graph-wide metrics).

It has to be observed that the *GEO SELECT* actor specifies an operation taking place on the client side, taking into account the preferences of the user (area of interest selected on the map). Instead, the *COMPUTE METRICS* actor is implemented on the server side.

Secondly, it is relevant to observe that GRRASP supports DFGs expressing computational operation starting from geographic data and resulting in non necessarily geographical data.

1.2 Module for the analysis of economic impact of critical infrastructure disruption

In this case the objective is the implementation in GRRASP of the Dynamic Input-Output Inoperability Model with inventories (I-DIIM) [12] with extensions. The objective is the assessment of the inoperability propagation through economic sectors as a consequence of a shock, taking into account the economical exchange between sectors described by input-output tables, the dynamics of the propagation phenomenon and the mitigating role of inventories to damp the effects of critical events.

Description of the computational process

In the current implementation, the analysis is performed on a national basis. Starting from yearly data available from the World Input-Output Database (`http://www.wiod.org/`) for different countries, the process is performed in GRRASP by incorporating the WIOD tables for different countries. Transformations are applied over these data in order to provide a portion of the parameters relevant to the I-DIIM model. The user completes the specification of the model parameters and simulation setup by selecting a country of reference and specifying, sector-by-sector, the initial inoperability for each sector, the duration of the inoperability before recovery starts, the duration of the recovery and the assumed inventory levels. The analysis is then performed on the server. The model provides a time inoperability graph for each sector, as well as the economic impact for each sector and the aggregated impact for all sectors.

DFG representation

The implementation chosen for the present module includes two key nodes, see Figure 2.2. These are the *COUNTRY SELECT* node and the *COMPUTE I-DIIM* node. The first of them is specified as a client-side operation performed by according to the user's preference. Instead the *COMPUTE I-DIIM* node involves a server-side computational component, triggered by the user under the specification of the residual parameters relevant to the simulation.

The data sources in this case are both geographical (countries for which WIOD tables are included in the module) and non-geographical (WIOD tables and their transformations as described above).

The output in this case is totally non-geographical, as it consists (for a specified country) in the inoperability time profiles relative to the different sectors involved in the

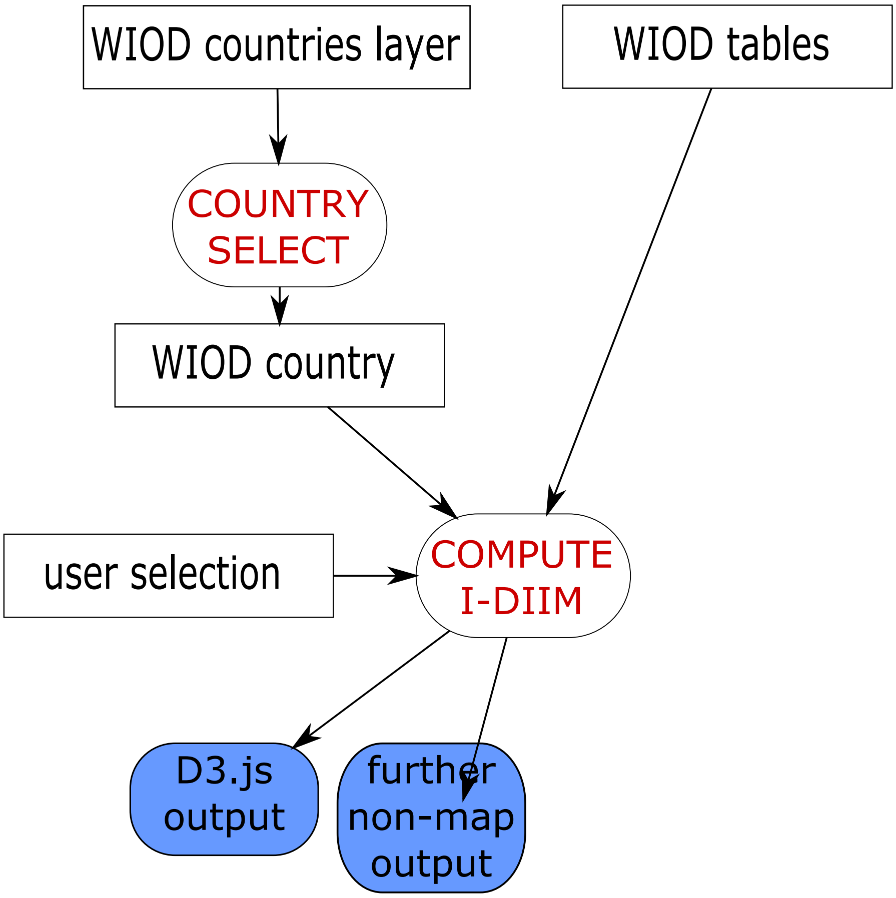

Figure 2.2: *DFG representation of the I-DIIM module.*

analysis plus some further output (e.g. aggregated losses).

2 Discussion

The implementation examples introduced above in this section emphasize a number of key aspects that are relevant to the DFG representation of scientific modules in GRRASP:

- **scientific modules as DFGs (with inputs)**: in the reported examples, the computation is performed by a combination of geo-based operations plus some interaction with the user, typically requested to complete the parameterization of the model and triggering the computation; the overall process of one such module, targeting the launch of a specific mathematical model or class of models, has the structure of a DFG with inputs;
- **hardware deployment**: in GRRASP, the server-client architecture allows the pos-

siblity to deploy parts of a given DFG on one or the other side; in the currently implemented modules, typically the server is loaded with the most computationally-intensive nodes of a specified DFG; different criteria can contribute to this choice in practice (e.g. privacy of some computational routines, availability of the system to tiny client devices);

- **etherogeneous I/O data types**: the data flow allowed in GRRASP is not completely geographical, which is a choice allowing to enlarge the pool of scientific modules compatible with this interface; as the system evolves, an effort is done to standardize the data structures supported and to allow the access to different visualization methods as a functions of data types;
- **DFGs sharing node types**: the *GEO SELECT* node of the network analysis DFG and the *COUNTRY SELECT* of the I-DIIM DFG share a similar objective, i.e. performing a geo-based filtering of data on the client side; more generally, different scientific modules share pieces of their DFGs (e.g. nodes) with others; this point is relevant to the formulation of development guidelines for GRRASP, as further discussed in Chapter 3.

3 Tiering of DFGs

Standardization of the data flow along DFGs associated to the scientific modules has also the objective to allow the combination of different DFGs to allow tiered analysis, by setting up potentially complex analysis routines expressed as larger DFGs, in themselves, thus incorporating and combining multiple modules.

As mentioned in the conclusions of previous reports, GRRASP aims at a tiered approach (see Figure 2.3) to the analysis of CIs. In particular, as illustrated in the figure, we we envision analysis processes that may encompass three tiers, namely sectoral analysis, cross-sectoral analysis and high-level service impact analysis. Next we discuss some aspects of the three.

Tier 1 (sectoral analysis). This tier constitutes the basis of most simulation software for critical infrastructure analysis and obviously there is a reason for this. Research institutes and scientists are often specialised in a particular domain and for this reason there is the tendency to develop detailed engineering models. Typically, such approaches require a high amount of specialised data. On the other hand, these models can provide very detailed descriptions of critical infrastructures and exhibit limited uncertainty, while they often require considerable development time. Further, typically they can only be used by experts in the respective field and the developers have certainly the primary ownership due to the inherent complexity of such systems. In principle the maturity in this area is high and the vast majority of actors in the field are focused on this particular Tier. In this Tier one may find models that are applicable at all levels (local, regional, national, international), however, their complexity and difficulty rather increases as we scale-up towards national/international level. An example of a model

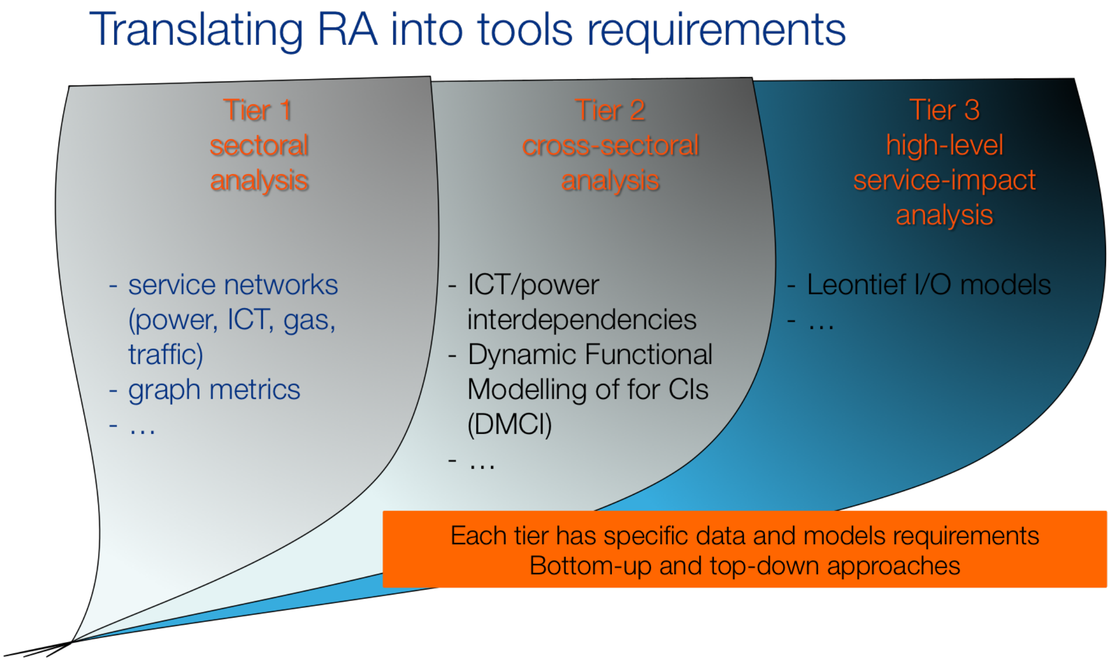

Figure 2.3: *Tiered CI analysis approach*

in GRRASP belonging to this tier is the Geomagnetically Induced Current module that evaluates the development of geomagnetically induced currents on power grids due to the variation of Earth's magnetic field that follows severe space weather events. Another example is the one of structural analysis of networks.

Tier 2 (cross-sectoral analysis). By definition, Tier 2 includes models that require more knowledge on the interactions between sectors and less specific knowledge on the particular dynamics of a sector. Piecing together models belonging to the first tier while addressing different sectors might lead one to think to obtain an analysis of interdependent systems however, this is not the case. Although this may seem reasonable as a claim, in reality it is strenuous due to the tremendous complexity that this approach would generate and also imply a request for a huge amount of data. So it is necessary to adopt a different approach that focuses on higher-level variables such as demand and delivery of services and in that way interdependent infrastructures can be modeled with less data and also reduced complexity. Here we have much fewer models, although their complexity can be even lower with respect to Tier 1 models. It is important to mention here that Tier 2 models are applicable at all levels but certainly their real strength is shown when it comes to regional and national level. At an international level it is very important to represent large parts of infrastructures with a limited amount of information otherwise there is the risk to go towards first tier models. Tier 2 modules are related to the assessment of interdependencies between sectors of critical infrastructures. Interdependencies can be classified as functional, logical, cyber and geographical and certainly a robust interdependencies analysis module should be able to take into account all these types of interdependencies. In order to address this issue we have jointly developed with Polytechnic School of Milan an interdependencies analysis module, the DMCI (Dynamic Functional Modelling of vulnerability and interoperability of CIs)1 that takes into account the above mentioned types of interdependencies while its modularity enables the end user to define nodes of critical infrastructures on a map and establish cross-sectoral interdependencies among these assets. Among other advantages, this type of tool enables the collaboration of multiple actors in the field thus it facilitates a bottom up approach towards improving the understanding of interdependencies among sectors. Relevant application examples include the impact assessment of power grid disruptions on telecommunications or the effects of a disruption in the rail transports on the road transport network due to the transfer of service demand by the end users.

Tier 3 (high-level service impact analysis). This tier focuses on the assessment of high level impact at regional, national and international level taking input from the modules of Tier 1 and Tier 2, where relevant. In this framework we can collocate the economic impact module that has been introduced in GRRASP and is based on an inoperability Input/Output model3. This module includes enhanced features in order to describe the dynamics of the recovery process, while taking into account the existence of inventory within certain economic sectors. However, more modules are needed that can address important issues such as regionalization of the effects of critical events. Although some of these issues can be addressed, at a first stage, with a Tier 1 module, in that case the output would not be as accurate since high order effects (interdependencies) could be omitted. GRRASP's open architecture allows third party users to enrich the modules portfolio to complement existing capabilities of GRRASP across tiers. Currently the integration of the various modules belonging to different tiers is under development. This will lead to a seamless risk and resilience assessment framework, starting from the assessment of threats at sectoral level leading to estimate interdependencies between sectors and finally reaching the assessment of the total economic impact. The inclusion of further types of impact analysis at Tier 3 is also under development. In addition to these functionalities, we have equipped GRRASP with the capability to fetch data from remote servers and use them for visualization purposes or for initiating a Risk/Resilience analysis. This functionality enables GRRASP users to set up dynamic and interactive processes for information exchange and sharing of risk maps as well as other geospatially related data. Currently such services are deployed only in a few cases.

CHAPTER 3

Development guidelines

In the previous chapter, based on a DFG interpretation of some scientific modules implemented in GRRASP, we pointed out some key aspects relevant to the future development of the platform:

- **management of DFG implementation and data flows**: this involves the management of the computational deployment related to different modules and the creation of standardized formats for the interaction between scientific modules and other modules and/or data sources;
- **existence of common routines**: there are some types of operation (e.g. geo-filtering and selection) which are common to multiple modules and therefore can be considered core functionalities of the platform.

One further point related to the management of the computational flow is related to the traceability of the processing underlying the data flowing across the system.

In this chapter we discuss how then above mentioned aspect affects the development process of the GRRASP platform

1 Core and library development

We can articulate the development of GRRASP in two aspects:

- **core development**: this is about the development of basic mechanisms of the server-client architecture, the interfacing with data sources and possibly different installations of the platform, plus routines related to the management of the DFG-based computation and shared DFG components, i.e. those of common use to multiple scientific modules;
- **library development**: this relates to the development of specialized components such as a scientific module.

Keeping such a distinction is also important since, depending on the objectives of the installation itself, a GRRASP installation may only comprise some libraries among those available. Furthermore, libraries may be contributed within a single installation only. Furthermore, distinct but correlated versioning mechanisms can also be put in place for the core software base and libraries.

2 Component migration

Migration of pieces of code from the libraries to the core base is a relevant process as GRRASP is being interfaced with different projects.

For instance, in the previous chapter we pointed out that different DFGs may share node types, as in the case of the *GEO SELECT* node for the network analysis DFG and the *COUNTRY SELECT* node of the I-DIIM DFG. This aspect emphasized the necessity for the GRRASP architecture to provide a set of selection tools able to support complex geometries and options such as the single- or multi-layer selections.

Another aspect that is being tackled is the automation of the data flow management processes, including the linkage between scientific outputs and visualization routines. The system supports the visualization of map outputs form scientific modules as well as non-map outputs. Both of them require the specification of visualization options. In the first case, a set of JavaScript routines was developed in order to allow a wide exploitation of the visualization capabilities of the OpenLayers library. In the second case, instead, flexibility is allowed and further JavaScript routines have been developed within the D3.js framework for scientific visualization purpose. In both cases, the redirection of model output to the relevant visualization routine can be managed through core components of the GRRASP architecture.

Finally, it has to be mentioned that, in time, the component migration process from libraries to the core of the GRRASP libraries can render the scientific module development process easier by the exploitation of out-of-the-box components provided by the GRRASP core architecture.

3 Perspective on core development

The core functionalities of the system are intended to integrate the key operational logics and data flow processing. In particular, we can enumerate the following aspects:

- **interfacing with and browsing external data sources**: these include internally provided map services such as the European Data Portal (http://www.europeandataportal.eu/), the US Government's data portal (https://www.data.gov/), the Center for International Earth Science Information Network of The Columbia University's Earth Institute (http://ciesin.columbia.edu/) (see also, for instance, http://www.skylab-mobilesystems.com/en/wms_serverlist.html for a lista of WMS services);

- **core processing functionalities**: filters, layer creation and drawing functionalities are being enhanced;
- **DFG management**: computation flow traceability is being addressed in order to allow attaching source and upstream processing information to to data flows; interfacing with dedicated DFG-based computational architectures is under evaluation;
- **DFG blueprinting**: the objective is to allow the user to setup complex analysis streams (e.g. tiered DFGs), by selecting and combining elementary DFGs where relevant.

CHAPTER 4

Project example

Recently, a collaboration was initiated between the JRC and the RAIN-EX project (http://www.rainex-project.eu/) towards the implementation of methodologies proposed within the project activities and GRRASP[1]

1 Basic information about the RAIN-EX project

The RAIN-EX project (*"Risk-Based Approach for the Protection of Land Transport Infrastructure against Extreme Rainfall"*) aims at ensuring the availability of transport infrastructure with regards toi natural hazards, especially extreme rainfall, through a risk-based design of the former.

The key project objectives are

- to advance the design of new land transport infrastructure with regard to security aspects;
- to develop a comprehensive approach for a risk-based assessment and adaptation of existing land transport infrastructure;
- to foster the awareness of land transport infrastructure owners and operators to ensure the availability of their network;
- to support the dissemination and implementation of the developed methodology via a user-friendly handbook.

(see http://www.rainex-project.eu/wp-content/uploads/RAIN-EX_1st_AB-Meeting_short_Presentation.pdf).

[1] *"The RAINEX consortium participated at the CIPS V Workshop held in Brussels on the 24 of February 2016, where the actual status of the project and the developed methodology were presented. On the 20th of April 2016 another meeting with the DG Joint Research Centre in Ispra, Italy, followed in order to discuss the implementation of the RAINEX methodology in the Geospatial Risk and Resilience Assessment Platform (GRRASP) of the European Commission. RAINEX will be the first of the CIPS Projects that will be integrated into this platform"* (from http://www.rainex-project.eu/publications/).

The project exploits a base of geo-information comprising digital terrain and landscape information, hazard zone maps and digital aerial images combined with meteorological data. By the uses of a series of climatic models, statistical and temporal analyses, emission scenarios, the proposed methodology aims at identifying which structures exposed to rainfall-induced natural hazards might possibly be of higher concern for protection.

2 Integration of RAIN-EX methodology in GRRASP

The integration of some of the outcomes of the RAIN-EX project in GRRASP in an ongoing initiative which provides an example o interaction of the GRRASP platform with existing methodologies for the analysis of critical infrastructures.

The process starts from three pools of data:

- **pool 1**: digital terrain model, digital landscape model, hazard zone maps; digital aerial images;
- **pool 2**: meteorological data;
- **pool 3**: infrastructure database, geological information, hydrological information, topological information, transport network information;

Upon the specification of preferences by the user (geographical preferences, type of hazard taken into consideration etc.) the calculation of exposure and vulnerability indexes is triggered. Examples of hazards taken into account in the study include riverine flooding, ponding, hillslope flooding, and debris flow. The next step is the processing of the exposure and vulnerability results to calculate risk according to the methodology proposed by the project. In Figure 4.1 we propose a DFD representation of the procedure described above.

The assessment process leads to the specification of likelihood-consequence matrices associated to the different infrastructures involved in the analysis and taking into account the considered hazards. An example of such tables is provided in Figure 4.2.

Optionally, the methodology also allows the formulation of a risk prognosis, see Figure 4.3.

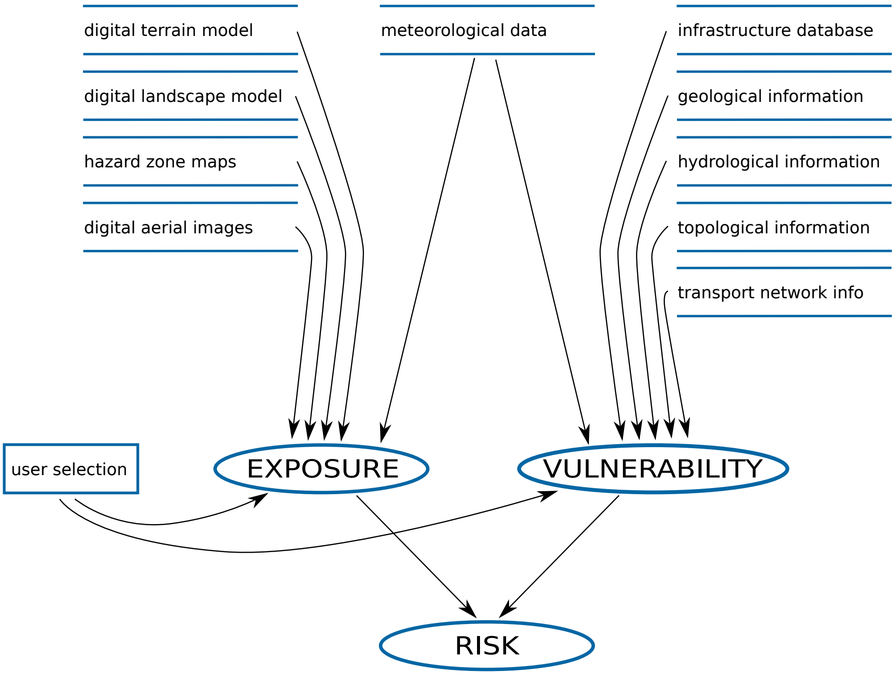

Figure 4.1: *DFG representation of the RAIN-EX methodology.*

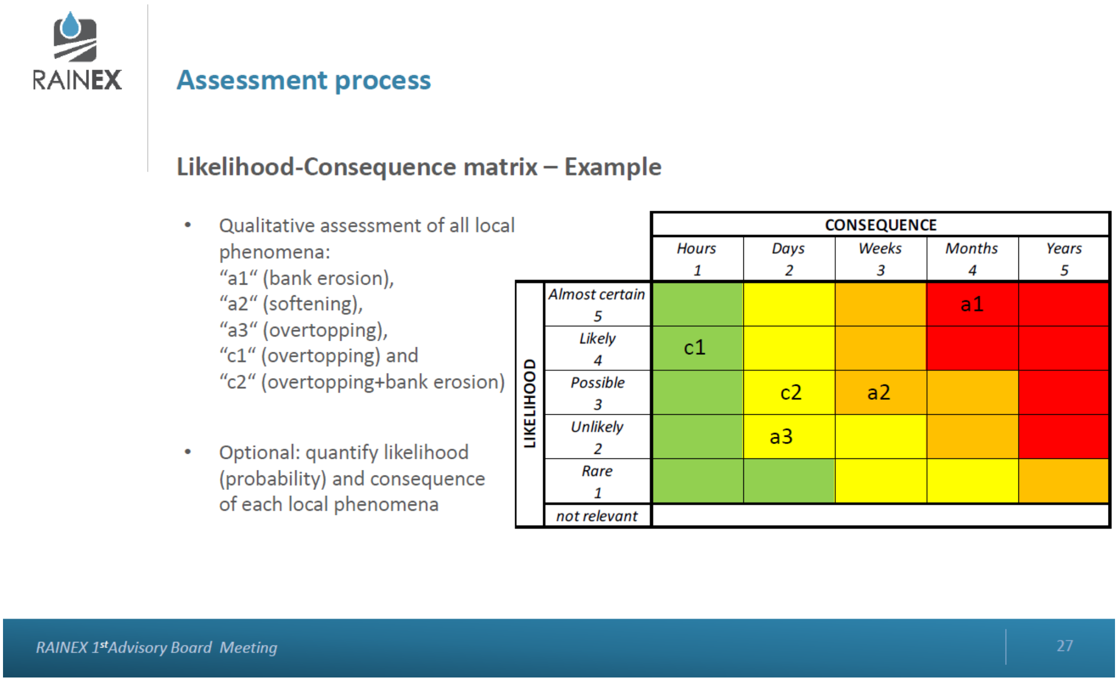

Figure 4.2: *RAIN-EX Project: likelihood-consequence matrices (http://www.rainex-project.eu/wp-content/uploads/RAIN-EX_1st_AB-Meeting_short_Presentation.pdf).*

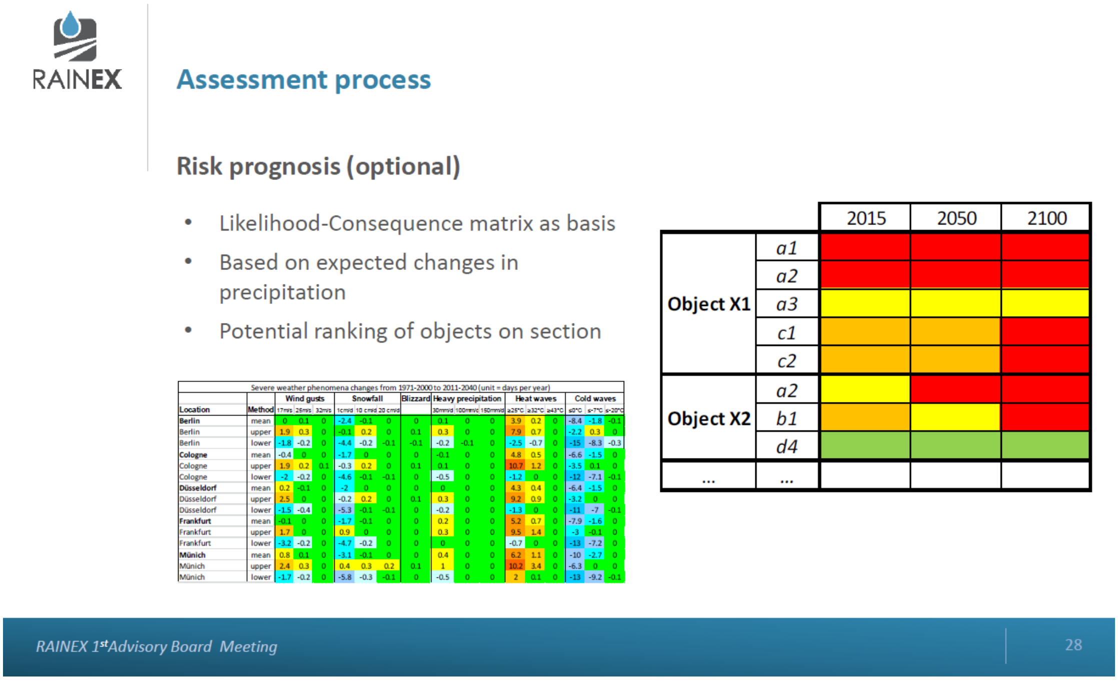

Figure 4.3: *RAIN-EX Project: risk prognosis (`http://www.rainex-project.eu/wp-content/uploads/RAIN-EX_1st_AB-Meeting_short_Presentation.pdf`).*

Conclusions

In this report we illustrated how the modular architecture of the new version of the GRRASP platform is able to integrate and support new components and to interface with projects related to CI analysis.

The objective was achieved by pointing out the key aspects of the computational architecture we are implementing in the development process. The data flow diagram formalism was exploited to illustrated our orientation towards a dataflow programming mechanism, which seems a sound choice thanks to its emphasis on graph-like processing, parallelism, support for large and decentralized deployment. Also, it could be interesting in order to interface GRRASP to such tools as Google's TensorFlow™, for instance in order to develop artificial intelligence-based geo-processing.

We explained how some scientific modules included in the current GRRASP version admit a DFG representation, which is also exploitable towards model tiering into larger analysis frameworks, starting from sectoral analysis routines and moving to cross-sectoral studies up to high level, service impact analysis.

Finally, we provided an example of integration of an existing CI analysis project in the platform. This was done with reference to the RAIN-EX project, addressing the protection of land transport infrastructure against extreme rainfall.

Bibliography

[1] M. Ouyang, “Review on modeling and simulation of interdependent critical infrastructure systems,” *Reliability engineering & System safety*, vol. 121, pp. 43–60, 2014.

[2] O. Jonkeren, I. Azzini, L. Galbusera, S. Ntalampiras, and G. Giannopoulos, “Analysis of critical infrastructure network failure in the european union: a combined systems engineering and economic model,” *Networks and Spatial Economics*, vol. 15, no. 2, pp. 253–270, 2015.

[3] W. M. Johnston, J. Hanna, and R. J. Millar, “Advances in dataflow programming languages,” *ACM Computing Surveys (CSUR)*, vol. 36, no. 1, pp. 1–34, 2004.

[4] J. Dennis, “Data flow graphs,” *Encyclopedia of Parallel Computing*, pp. 512–518, 2011.

[5] W. B. Ackerman, “Data flow languages,” *Computer*, vol. 15, no. 2, pp. 15–25, 1982.

[6] A. Davis and R. Keller, “Data flow program graphs,” *Computer*, vol. 15, no. 2, pp. 26–41, 1982.

[7] M. Marttila-Kontio, “Visual data flow programming languages,” Ph.D. dissertation, University of Eastern Finland, 2011.

[8] G. G. de Jong, “Verification of dataflow graphs using temporal logic,” *Proc. IMEC-IFIP WG*, vol. 10, no. 10.5, pp. 301–310, 1989.

[9] P. R. Schaumont, “Data flow modeling and transformation,” in *A Practical Introduction to Hardware/Software Codesign*. Springer, 2013, pp. 31–59.

[10] R. G. Fichman and C. F. Kemerer, “Object-oriented and conventional analysis and design methodologies,” *Computer*, vol. 25, no. 10, pp. 22–39, 1992.

[11] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, “Tensorflow: Large-scale machine learning on heterogeneous distributed systems,” *arXiv preprint arXiv:1603.04467*, 2016.

[12] K. Barker and J. R. Santos, “Measuring the efficacy of inventory with a dynamic input–output model,” *International Journal of Production Economics*, vol. 126, no. 1, pp. 130–143, 2010.

Europe Direct is a service to help you find answers to your questions about the European Union
Freephone number (*): 00 800 6 7 8 9 10 11
(*) Certain mobile telephone operators do not allow access to 00 800 numbers or these calls may be billed.

A great deal of additional information on the European Union is available on the Internet.
It can be accessed through the Europa server *http://europa.eu*.

How to obtain EU publications

Our publications are available from EU Bookshop *(http://bookshop.europa.eu)*,
where you can place an order with the sales agent of your choice.

The Publications Office has a worldwide network of sales agents.
You can obtain their contact details by sending a fax to (352) 29 29-42758.

European Commission
JRC 102547 EN – Joint Research Centre –**Institute for Protection and Security of the Citizen**

Title: **Interconnecting GRRASP with additional platforms and tools: A feasibility study**
Author(s): Luca Galbusera, Georgios Giannopoulos

Luxembourg: Publications Office of the European Union

2016 – 36 pp. – 21.0 x 29.7 cm

LB-NA-28073-EN-N

JRC Mission

As the Commission's in-house science service, the Joint Research Centre's mission is to provide EU policies with independent, evidence-based scientific and technical support throughout the whole policy cycle.

Working in close cooperation with policy Directorates-General, the JRC addresses key societal challenges while stimulating innovation through developing new methods, tools and standards, and sharing its know-how with the Member States, the scientific community and international partners.

Serving society
Stimulating innovation
Supporting legislation

Publications Office

doi:10.2788/9389

ISBN 978-92-79-61295-4